UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/720,969	11/24/2003	Haim Kopylovitz	EMS-06901	5218

7590 10/16/2006

Patent Group
Choate, Hall & Stewart
Exchange Place
53 State Street
Boston, MA 02109-2804

EXAMINER	
LOVEL, KIMBERLY M	

ART UNIT	PAPER NUMBER
2167	

DATE MAILED: 10/16/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/720,969	KOPYLOVITZ, HAIM	
	Examiner	Art Unit	
	Kimberly Lovel	2167	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *04 August 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-18* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-18* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This communication is responsive to the Amendment filed 4 August 2006.

2. Claims 1-18 are pending in this application. Claims 1 and 10 are independent. In the Amendment filed 4 August 2006, claims 1, 8 and 10 have been amended. This action is made Final.

3. The rejections of claims 1-7 and 10-16 as being unpatentable over the article "File System Design for an NFS File Server" by Hitz et al in view of Patent No. 5,819,292 to Hitz et al and claims 8-9 and 17-18 as being unpatentable over the article "File System Design for an NFS File Server" by Hitz et al in view of Patent No. 5,819,292 to Hitz et al in view of US Patent No. 6,460,054 to Grummon have been withdrawn as necessitated by the amendment.

Specification

4. The objection to the abstract of the disclosure is withdrawn as necessitated by the amendment.

Claim Rejections - 35 USC § 101

5. The rejections of claims 1-18 under 35 U.S.C. 101 are withdrawn as necessitated by the amendment.

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. Claims 1-9 are rejected under 35 U.S.C. 103(a) as being unpatentable over the article "File System Design for an NFS File Server" by Hitz et al (hereafter File System Design) in view of Patent No. 5,819,292 to Hitz et al (hereafter '292) in view of the article "A Persistent Snapshot Device Driver for Linux" by Siddha (hereafter Siddha) in view of the background of US Patent No. 6,460,054 to Grummon (hereafter Grummon).

Referring to claim 1, File System Design discloses a method for restoring data. In particular, File System Design discloses a method of restoring data (see section 2: Introduction to Snapshots, lines 9-12) to a first storage device, comprising:

providing data in a first storage area of a first type that contains sections of data (see section 3.4: Snapshots, lines 1-4 and 13-20 and Fig 3c, items A,B,C,D,E);

providing data in a second storage area of a second type wherein the second type has, for each section of data thereof (see Fig 3c – the root inode and New Snapshot are of the second *data type*), at least one of:

a pointer to a corresponding section of data of the first storage area and pointer to corresponding section of data of a third storage area of the first type (see section 3.4: Snapshots, lines 10-12);

providing data in having at least one other storage area of the second type (see Fig 3c – the third storage area contains the changed data D' and the root node of Fig. 3c contains pointers to the first storage area (A,B,C,D,E) and a pointer to the third storage area containing D'); and

for each particular section of data of the second storage area having a pointer to the third storage area (see Fig. 3c – New Snapshot), providing to a corresponding section of the first storage area an indirect pointer to a corresponding section of the third storage area if no storage areas of the at least one other storage area point to the corresponding section of the first storage area.

While File System Design teaches a pointer to the third storage area for each particular section of data of the second storage area, File System Design fails to explicitly teach the further limitations of wherein prior to writing new data to a section of the first storage area pointed to by a pointer of the second storage area, data of the section of the first storage area is copied to a section of the third storage area and the pointer of the second storage area is adjusted to point to the section of the third storage area and providing to a corresponding section of the first storage area an indirect pointer to a corresponding section of the third storage area if no storage areas of the at least one other storage area point to the corresponding section of the first storage area. '292 discloses a method for restoring data similar to that of File System Design

including the further limitations of wherein prior to writing new data to a section of the first storage area pointed to by a pointer of the second storage area, data of the section of the first storage area is copied to a section of the third storage area and the pointer of the second storage area is adjusted to point to the section of the third storage area (see Fig 18C and corresponding Fig 19) and providing to a corresponding section of the first storage area an indirect pointer to a corresponding section of the third storage area if no storage areas of the at least one other storage area point to the corresponding section of the first storage area (see column 18, line 49 through column 19, line 50 and Figures 18A-18C).

It would have been obvious to one of ordinary skill in the art at the time of the invention to utilize the indirect pointers of '292 as a subcomponent of the method for restoring data. One would have been motivated to do so since both are directed towards maintaining consistent states of a file system ('292: see abstract; File System Design: see abstract) and are written by common authors.

The combination of Design and '292 (hereafter Design/'292) fails to explicitly disclose wherein the new data is written to the section of stored data. However, Siddha et al discloses a method wherein the new data is written to the section of the stored data (see page 1, col. 2, second and third paragraphs – creating a snapshot and instituting copy-on-write technique, wherein the contents of blocks that are to be modified are copied to the snapshot save area, and after the block is copied, its physical location can be overwritten by the changed data) since this technique would have smaller performance impact than alternative online backup approaches.

It would have been obvious to one of ordinary skill in the art at the time of the invention to write new data to the existing data. One would have been motivated to do so since this technique would have smaller performance impact than alternative online backup approaches (Siddha et al: see page 1, col. 2, third paragraph).

The combination of Design/'292 and Siddha (hereafter Design/'292/Siddha) fails to explicitly disclose the limitation wherein the storage areas are devices. Grummon teaches Computer software similar to that of Design/'292/Siddha for restoring data using snapshots. In particular, Grummon teaches restoring data using snapshots, including the further limitation, wherein the storage areas are devices (see column 1, lines 29-59) in order to allow management of data at a low (logical volume/disk formatting) level, thus allowing efficient storage of data on physical media.

It would have been obvious to one of ordinary skill in the art at the time of the invention to implement the claimed software wherein each storage area is a storage device. One would have been motivated to do so to allow management of data at a low (logical volume/disk formatting) level, thus allowing efficient storage of data on physical media.

Referring to claim 2, the combination of Design/'292/Siddha and Grummon (hereafter Design/'292/Siddha/Grummon) discloses a method, according to claim 1, further comprising:

for each particular section of data of the second storage area having a pointer to the third storage area, providing to a corresponding section of the first storage area a doubly indirect pointer to a corresponding section of the third storage area if the at least

one other storage area points to the corresponding section of the first storage area ('292: see column 8, lines 39-55 – occurs when the file size is greater than 64MB).

Referring to claim 3, Design/'292/Siddha/Grummon discloses a method, according to claim 2, further comprising:

causing data to be copied from the third storage area to the first storage area for each section of the first area having associated therewith one of: an indirect pointer and a doubly indirect pointer ('292: see column 9, lines 25-48).

Referring to claim 4, Design/'292/Siddha/Grummon discloses a method, according to claim 3, further comprising:

in response to a particular section of the first storage area having associated therewith a doubly indirect pointer, copying data from the particular section of the first storage area to a new section of the third storage area prior to causing data to be copied to the particular section of the first storage area ('292: see column 9, lines 25-48).

Referring to claim 5, Design/'292/Siddha/Grummon discloses a method, according to claim 1, further comprising:

prior to replacing a corresponding section of the first storage area, disabling access to the first storage area and the second storage area ('292: see column 12, lines 39-47).

Referring to claim 6, Design/'292/Siddha/Grummon discloses a method, according to claim 5, further comprising:

after replacing a corresponding section of the first storage area for all of the particular sections of data of the second storage area having a pointer to the third storage area, enabling read and write access to the first storage area and enabling read access to the second storage area ('292: see column 12, lines 43-45 – after the consistency flag is lifted, then read and write access is enabled).

Referring to claim 7, Design/'292/Siddha/Grummon discloses a method, according to claim 5, further comprising:

after replacing a corresponding section of the first storage area for all of the particular sections of data of the second storage area having a pointer to the third storage area, enabling read and write access to the first and second storage areas ('292: see column 12, line 48 – column 13, line 2 – after the global consistency flag is lifted, then read and writes can occur).

Referring to claim 8, Design/'292/Siddha/Grummon discloses wherein the first storage device, the second storage device, the third storage device, and the fourth storage device are provided by a single physical storage device [volume] (see column 1, lines 29-59).

Referring to claim 9, Design/'292/Siddha/Grummon discloses a method, according to claim 8, wherein the sections are tracks (Grummon: see column 1, lines 29-59).

8. Claims 10-16 are rejected under 35 U.S.C. 103(a) as being unpatentable over the article "File System Design for an NFS File Server" by Hitz et al in view of Patent No. 5,819,292 to Hitz et al in view of the article "A Persistent Snapshot Device Driver for Linux" by Siddha.

Referring to claim 10, File System Design discloses a method for restoring data. In particular, File System Design discloses computer software, provided in a computer-readable storage medium, that restores data to a first storage area of a first type that contains sections of data (see section 3.4: Snapshots, lines 1-4 and 13-20 and Fig 3c, items A,B,C,D,E) from a second storage area of a second type (see Fig 3c – the root inode and New Snapshot are of the second *data type*) that has, for each section of data thereof, at least one of:

a pointer to a corresponding section of data of the first storage area and a pointer to corresponding section of data of a third storage area of the first type where there is at least one other storage area of the second type (see section 3.4: Snapshots, lines 10-12), the software comprising:

executable code that iterates through each section of the second storage area (see section 3.4: Snapshots, lines 34-43 and Fig 4); and

executable code that provides to a corresponding section of the first storage area an indirect pointer to a corresponding section of the third storage area if no storage areas of the at least one other storage area point to the corresponding section of the first storage area.

While File System Design teaches a pointer to the third storage area for each particular section of data of the second storage area, File System Design fails to explicitly teach the further limitations executable code that, prior to writing new data to a section of the first storage area pointed to by a pointer of the second storage area, data of the section of the first storage area is copied to a section of the third storage area and the pointer of the second storage area is adjusted to point to the section of the third storage area and of providing to a corresponding section of the first storage area an indirect pointer to a corresponding section of the third storage area if no storage areas of the at least one other storage area point to the corresponding section of the first storage area. '292 discloses a method for restoring data similar to that of File System Design including the further limitations of executable code that, prior to writing new data to a section of the first storage area pointed to by a pointer of the second storage area, data of the section of the first storage area is copied to a section of the third storage area and the pointer of the second storage area is adjusted to point to the section of the third storage area (see Fig 18C and corresponding Fig 19) and providing to a corresponding section of the first storage area an indirect pointer to a corresponding section of the third storage area if no storage areas of the at least one other storage area point to the corresponding section of the first storage area (see column 18, line 49 through column 19, line 50 and Figures 18A-18C).

It would have been obvious to one of ordinary skill in the art at the time of the invention to utilize the indirect pointers of '292 as a subcomponent of the method for restoring data. One would have been motivated to do so since both are directed

towards maintaining consistent states of a file system ('292: see abstract; File System Design: see abstract) and are written by common authors.

Design/'292 fails to explicitly disclose wherein the new data is written to the section of stored data. However, Siddha et al discloses a method wherein the new data is written to the section of the stored data (see page 1, col. 2, second and third paragraphs – creating a snapshot and instituting copy-on-write technique, wherein the contents of blocks that are to be modified are copied to the snapshot save area, and after the block is copied, its physical location can be overwritten by the changed data).

It would have been obvious to one of ordinary skill in the art at the time of the invention to write new data to the existing data. One would have been motivated to do so since this technique would have smaller performance impact than alternative online backup approaches (Siddha et al: see page 1, col. 2, third paragraph).

Referring to claim 11, Design/'292/Siddha discloses Computer software, according to claim 10, further comprising:

executable code that provides to a corresponding section of the first storage area a doubly indirect pointer to a corresponding section of the third storage area if the at least one other storage area points to the corresponding section of the first storage area ('298: see column 8, lines 39-55 – occurs when the file size is greater than 64 MB).

Referring to claim 12, Design/'292/Siddha discloses Computer software, according to claim 11, further comprising:

executable code that causes data to be copied from the third storage area to the first storage area for each section of the first area having associated therewith one of: an indirect pointer and a doubly indirect pointer ('292: see column 9, lines 25-48).

Referring to claim 13, Design/'292/Siddha discloses Computer software, according to claim 12, further comprising:

executable code that copies data from the particular section of the first storage area to a new section of the third storage area prior to causing data to be copied to the particular section of the first storage area in response to a particular section of the first storage area having associated therewith a doubly indirect pointer ('292: see column 9, lines 25-48).

Referring to claim 14, Design/'292/Siddha discloses Computer software, according to claim 10, further comprising:

executable code that disables access to the first storage area and the second storage area prior to replacing a corresponding section of the first storage area ('292: see column 12, lines 39-47).

Referring to claim 15, Design/'292/Siddha discloses Computer software, according to claim 14, further comprising:

executable code that enables read and write access to the first storage area and enabling read access to the second storage area after replacing a corresponding section of the first storage area for all of the particular sections of data of the second storage area having a pointer to the third storage area ('292: see column 12, lines 43-45 – after the consistency flag is lifted, then read and write access is enabled).

Referring to claim 16, Design/'292/Siddha discloses Computer software, according to claim 14, further comprising:

executable code that enables read and write access to the first and second storage areas after replacing a corresponding section of the first storage area for all of the particular sections of data of the second storage area having a pointer to the third storage area ('292: see column 12, line 48 – column 13, line 2 – after the global consistency flag is lifted, then read and write access can occur).

9. Claims 17-18 are rejected under 35 U.S.C. 103(a) as being unpatentable over the article "File System Design for an NFS File Server" by Hitz et al in view of Patent No. 5,819,292 to Hitz et al in view of in view of the article "A Persistent Snapshot Device Driver for Linux" by Siddha (hereafter Siddha) as applied to claim 10 above, and further in view of the background US Patent No. 6,460,054 to Grummon (hereafter Grummon).

Referring to claim 17, Design/'292/Siddha discloses Computer software for restoring data using snapshots. However, Design/'292/Siddha fails to explicitly teach the further limitation wherein the storage areas are devices. Grummon teaches restoring data using snapshots, including the further limitation, wherein the storage areas are devices (see column 1, lines 29-59) in order to allow management of data at a low (logical volume/disk formatting) level, thus allowing efficient storage of data on physical media.

It would have been obvious to one of ordinary skill in the art at the time of the invention to implement the claimed software wherein each storage area is a storage device. One would have been motivated to do so to allow management of data at a low (logical volume/disk formatting) level, thus allowing efficient storage of data on physical media.

Referring to claim 18, the combination of Design/'292/Siddha/Grummon discloses computer software, according to claim 17, wherein the sections are tracks (Grummon: see column 1, lines 29-59).

Response to Arguments

10. Applicant's arguments with respect to claim 1-18 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

11. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of

the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Contact Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Kimberly Lovel whose telephone number is (571) 272-2750. The examiner can normally be reached on 8:00 - 4:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Cottingham can be reached on (571) 272-7079. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Kimberly Lovel
Examiner
Art Unit 2167

11 October 2006
kml

KML
12 October 2006

JOHN COTTINGHAM
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

12 October 2006